

中山大学移动信息工程学院本科生实验报告

(2017 年秋季学期)

课程名称：移动应用开发

任课教师： 郑桂锋

年级	15	专业 (方向)	移动
学号	15352403	姓名	张耕维
电话	15626176071	Email	zhanggw8@mail2.sysu.edu.cn
开始日期	2017-9-26	完成日期	2017-9-26

一、 实验题目

实验一 基本 UI 界面设计

二、 实验内容

使用 XML 实现一个基本界面，为启动后看到的第一个界面，使用 Constraintlayout 实现：

1. 标题字体大小 20sp，与顶部距离 20dp，居中；
2. 图片与标题的间距为 20dp，居中；
3. 输入框整体距屏幕右边间距 20dp，上下两栏间距 20dp，内容（包括提示内容）如图所示，内容字体大小 18sp
4. 学号对应的 EditText 只能输入数字，密码对应的 EditText 输入方式为密码；
5. 两个单选按钮整体居中，字体大小 18sp，间距 10dp，默认选中的按钮为第一个；
6. 两个按钮整体居中，与上方控件间距 20dp，按钮间的间距 10dp，文字大小 18sp。按钮背景框左右边框与文字间距 10dp，上下边框与文字间距 5dp，圆角半径 10dp，背景色为 #3F51B5

三、 课堂实验结果

(1) 实验截图

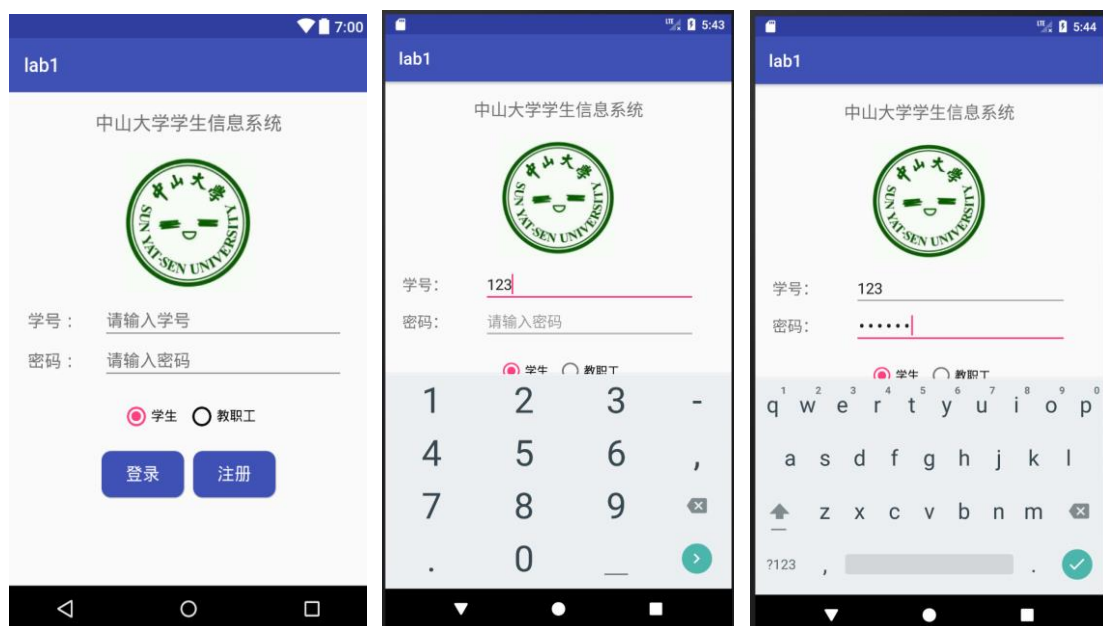


图 1 界面实现结果

(2) 实验步骤以及关键代码

a) 标题字体大小 20sp，与顶部距离 20dp，居中；

使用 guideline 确定距离顶部距离为 20dp，通过左右的 constraint 居中

```
android:textSize="20sp"
app:layout_constraintTop_toBottomOf="@id/title_hor_guideline"
app:layout_constraintLeft_toLeftOf="parent"
app:layout_constraintRight_toRightOf="parent" />
```

b) 图片与标题的间距为 20dp，居中；

使用 marginTop 控制图片与标题之间距离为 20dp

```
android:src="@mipmap/sysu"
android:contentDescription="中山大学学生信息系统"
app:layout_constraintLeft_toLeftOf="parent"
app:layout_constraintRight_toRightOf="parent"
app:layout_constraintTop_toBottomOf="@id/title"
android:layout_marginTop="20dp" />
```

c) 输入框整体距屏幕右边间距 20dp，上下两栏间距 20dp，内容（包括提示

内容) 如图所示, 内容字体大小 18sp

```
android:text="学号:"  
android:textSize="18sp"  
app:layout_constraintTop_toBottomOf="@id/img"  
app:layout_constraintLeft_toLeftOf="parent"  
android:layout_marginLeft="20dp"  
android:layout_marginTop="20dp"
```

同一行间距用一个控件控制即可, 设置 `ems` 控制输入框长度, 使用 `inputType` 指定输入类型

```
android:textSize="18sp"  
android:inputType="number"  
android:hint="请输入学号"  
android:ems="12"  
app:layout_constraintTop_toTopOf="@id/id_num"  
app:layout_constraintBottom_toBottomOf="@id/id_num"  
app:layout_constraintLeft_toRightOf="@id/id_num"  
app:layout_constraintRight_toRightOf="parent"
```

- d) 两个单选按钮整体居中, 字体大小 18sp, 间距 10dp, 默认选中的按钮为第一个;

使用 `radioGroup` 实现多个单选按钮

```
<RadioButton  
    android:checked="true"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="学生"  
>
```

```
<RadioButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="教职工"
    app:layout_constraintRight_toRightOf="parent"
    android:layout_marginLeft="10dp"
/>
```

- e) 两个按钮整体居中，与上方控件间距 20dp，按钮间的间距 10dp，文字大小 18sp。按钮背景框左右边框与文字间距 10dp，上下边框与文字间距 5dp，圆角半径 10dp，背景色为#3F51B5

使用一个垂直方向的 guideline 来控制两个按钮之间的间距

```
android:textColor="@color/white"
android:layout_marginEnd="5dp"
android:layout_marginRight="5dp"
android:background="@drawable/shape"
android:layout_marginTop="20dp"
app:layout_constraintTop_toBottomOf="@id/radio"
app:layout_constraintEnd_toStartOf="@id/ver_guideline"
app:layout_constraintHorizontal_chainStyle="spread" />
```

```
android:text="注册"
android:textSize="18sp"
android:textColor="@color/white"
android:layout_marginTop="20dp"
android:layout_marginStart="5dp"
android:layout_marginLeft="5dp"
android:background="@drawable/shape"
app:layout_constraintTop_toBottomOf="@id/radio"
```

使用 drawable/shape 来构造按钮的形状，边框

```

<corners
    android:radius="10dp"/>

<solid
    android:color="#3F51B5"/>

<padding
    android:top="5dp"
    android:bottom="5dp"
    android:left="10dp"
    android:right="10dp"/>

```

(3) 实验遇到的困难以及解决思路

实现两个按钮居中并间距 10dp 的时候，要么无法控制间距，要么无法居中，后来使用了 guideline 控制了间距并保证了居中

四、 课后实验结果

(1) 实验截图

实现 输入学号 (以及密码) 后，点击 “登录” 按钮，切换到欢迎界面，显示 welcome 以及 学号

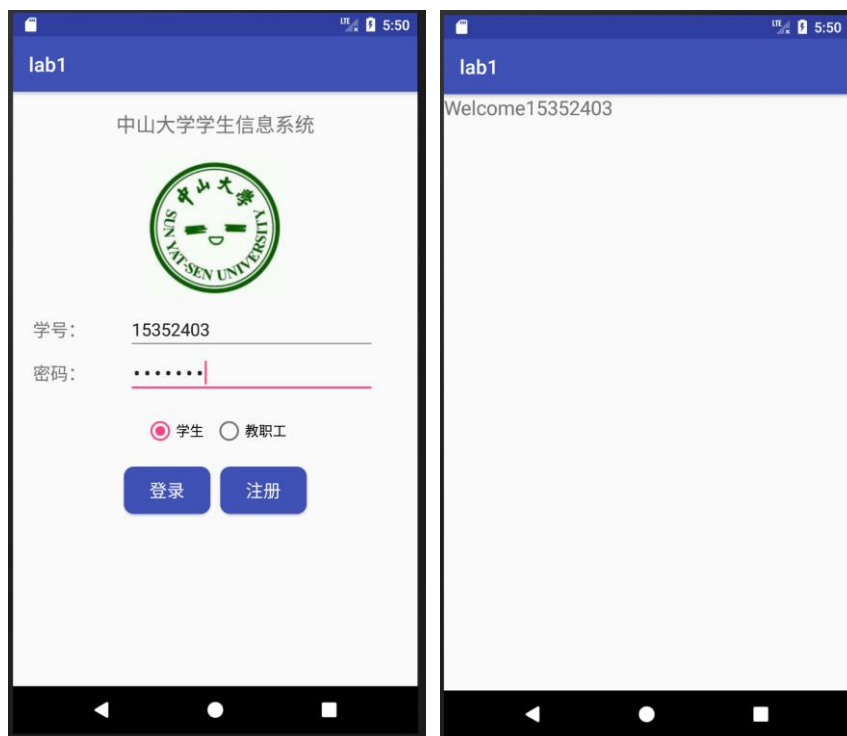


图 2 课后实验结果

(2) 实验步骤及关键代码

将输入的学号（通过 id）使用 intent 发送给另一个界面（activity）

```
public void sendMessage(View view) {  
    Intent intent = new Intent(this, MainActivity.class);  
    EditText editText = (EditText) findViewById(R.id.edt_num);  
    String message = editText.getText().toString();  
    intent.putExtra(EXTRA_MESSAGE, message);  
    startActivity(intent);  
}
```

在另一个 activity 接收 intent 并显示 welcome 和输入的学号

```
@Override  
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main2);  
  
    Intent intent = getIntent();  
    String message = intent.getStringExtra(MainActivity.EXTRA_MESSAGE);  
    TextView textView = new TextView(this);  
    textView.setTextSize(20);  
    textView.setText("Welcome"+message);  
  
    ViewGroup layout = (ViewGroup) findViewById(R.id.activity_display_message);  
    layout.addView(textView);  
}
```

五、 实验思考及感想

本次实验主要学习和练习了 constraintLayout 布局实现界面编程。ConstraintLayout 中的控件位置是根据布局中的其他元素或视图确定的，可以受父容器、其他控件、guideline 约束，因此能够帮助我们更好的布局。因此，灵活使用约束才能实现丰富的布局。例如在 constraintLayout 中，我们可以使用链样式排列元素，又能通过 guideline 约束控件，并且我们可以调整链样式中的权重，合理使用约束才能实现不同的布局。

课后又练习了 intent 的用法，并加入点击登录切换到欢迎界面的功能。